Pattern Recognition for Neuroimaging Data

Edinburgh, SPM course
April 2013

C. Phillips, Cyclotron Research Centre, ULg, Belgium
http://www.cyclotron.ulg.ac.be
Overview

- Introduction
  - Univariate & multivariate approaches
  - Data representation

- Pattern Recognition
  - Machine learning
  - Validation & inference
  - Weight maps & feature selection
  - fMRI application
  - Multiclass problem

- Conclusion & PRoNTo
Overview

• Introduction
  – Univariate & multivariate approaches
  – Data representation

• Pattern Recognition
  – Machine learning
  – Validation & inference
  – Weight maps & feature selection
  – fMRI application
  – Multiclass problem

• Conclusion & PRoNTo
Introduction

fMRI time series = 4D image
= time series of 3D fMRI’s
= 3D array of time series.
Univariate vs. multivariate

Standard univariate approach (SPM)

Find the mapping $g$ from explanatory variable $X$ to observed data $Y$

$$g: X \rightarrow Y$$
Univariate vs. multivariate

Multivariate approach, aka. “pattern recognition”

Find the mapping $h$ from observed data $Y$ to explanatory variable $X$

$h: Y \rightarrow X$
Neuroimaging data

3D brain image

“feature vector” or “data point”

Data dimensions

• dimensionality of a “data point” = #voxels considered
• number of “data point” = #scans/images considered

Note that #voxels >> #scans !

⇒ “ill posed problem”
Advantages of pattern recognition

Accounts for the spatial correlation of the data (multivariate aspect)

- images are multivariate by nature.
- can yield greater sensitivity than conventional (univariate) analysis.

Enable classification/prediction of individual subjects

- ‘Mind-reading’ or decoding
- Clinical application

Haynes & Rees, 2006
Pattern recognition framework

Input (brain scans)
\[ X_1 \quad X_2 \quad X_3 \]

No mathematical model available

Output (control/patient)
\[ y_1 \quad y_2 \quad y_3 \]

Machine Learning Methodology

Computer-based procedures that learn a function from a series of examples

Learning/Training Phase
Generate a function or classifier \( f \) such that
\[ f(x_i) \rightarrow y_i \]

Test Example \( X_i \)

Testing Phase
Prediction
\[ f(X_i) = y_i \]

Training Examples:
\[ (X_1, y_1), \ldots, (X_s, y_s) \]
Overview

• Introduction
  – Univariate & multivariate approaches
  – Data representation

• Pattern Recognition
  – Machine learning
  – Validation & inference
  – Weight maps & feature selection
  – fMRI application
  – Multiclass problem

• Conclusion & PRoNTo
Classification example

Different classifiers will compute different hyperplanes!

Note: task1/2 ~ disease/controle
Problem: 1000’s of features vs. 10’s of data points

Possible solutions to dimensionality problem:

- Feature selection strategies (e.g. ROIS, select only activated voxels)
- (Searchlight)
- **Kernel Methods**
Kernel approaches

- Mathematical trick! ➔ powerful and unified framework (e.g. classification & regression)

- Consist of two parts:
  - build the kernel matrix (mapping into the feature space)
  - train using the kernel matrix (designed to discover linear patterns in the feature space)

- Advantages:
  - computational shortcut ➔ represent linear patterns efficiently in high dimensional space.
  - Using the dual representation with proper regularization ➔ efficient solution of ill-conditioned problems.

- Examples ➔ Support Vector Machine (SVM), Gaussian Processes (GP), Kernel Ridge Regression (KRR),...
The “kernel function”

- 2 patterns $\mathbf{x}$ and $\mathbf{x}^*$ $\Rightarrow$ a real number characterizing their similarity (≈ distance measure).
- simple similarity measure = a dot product $\Rightarrow$ linear kernel.
Linear classifier

• hyperplanes through the feature space

• parameterized by
  – a weight vector $\mathbf{w}$ and
  – a bias term $b$.

• weight vector $\mathbf{w} = \text{linear combination of training examples } \mathbf{x}_i$ (where $i = 1,...,N$ and $N$ is the number of training examples)

$$\mathbf{w} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i$$

→ Find the $\alpha_i$ !!!
Linear classifier prediction

General equation: making predictions for a test example $\mathbf{x}_*$ with kernel methods

Primal representation

$$f(\mathbf{x}_*) = \mathbf{w} \times \mathbf{x}_* + b$$

Dual representation

$$f(\mathbf{x}_*) = \sum_{i=1}^{N} a_i \mathbf{x}_i \times \mathbf{x}_* + b$$

Kernel definition

$$f(\mathbf{x}_*) = \sum_{i=1}^{N} a_i K(\mathbf{x}_i, \mathbf{x}_*) + b$$

$f(\mathbf{x}_*) =$

- signed distance to boundary (classification)
- predicted score (regression)
**Support Vector Machine**

SVM = “maximum margin” classifier

Data: \( \{x_i, y_i\} \), \( i = 1, \ldots, N \)

Observations: \( x_i \in \mathbb{R}^d \)

Labels: \( y_i \in \{-1, +1\} \)

**Support vectors have** \( \alpha_i \neq 0 \)

\[
\mathbf{w} = \sum_{i=1}^{N} y_i \mathbf{x}_i
\]

\[
(w^T \mathbf{x}_i + b) = -1
\]

\[
(w^T \mathbf{x}_i + b) = +1
\]

\[
(w^T \mathbf{x}_i + b) < 0
\]

\[
(w^T \mathbf{x}_i + b) > 0
\]
Illustrative example: Classifiers as decision functions

Examples of class 1

Examples of class 2

New example

Weight vector or Discrimination map

Training

Testing

\[ f(x) = (w_1 \cdot v_1 + w_2 \cdot v_2) + b \]

\[ = (+5 \cdot 0.5 - 3 \cdot 0.8) + 0 \]

\[ = 0.1 \]

Positive value \( \Rightarrow \) Class 1
SVM vs. GP

SVM

- Hard binary classification
  - simple & efficient, quick calculation but
  - NO ‘grading’ in output \{-1, 1\}

Gaussian Processes

- probabilistic model
  - more complicated, slower calculation but
  - returns a probability \([0, 1]\)
  - can be multiclass
Overview

• Introduction
  – Univariate & multivariate approaches
  – Data representation

• Pattern Recognition
  – Machine learning
  – Validation & inference
  – Weight maps & feature selection
  – fMRI application
  – Multiclass problem

• Conclusion & PRoNTo
## Validation principle

### Samples

<table>
<thead>
<tr>
<th>Training set</th>
<th>Test set</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Table of samples" /></td>
<td><img src="image" alt="Table of samples" /></td>
</tr>
</tbody>
</table>

#### Variables:
- \( \text{var 1} \)
- \( \text{var 2} \)
- \( \text{var 3} \)
- ...  
- \( \text{var } m \)

#### Labeling:
- \( i \)
- \( i+1 \)
- \( i+2 \)
- ...  
- \( n \)

#### Accuracy evaluation:
- True label
- Predicted label
M-fold cross-validation

- Split data in 2 sets: “train” & “test”
  - evaluation on 1 “fold”
- Rotate partition and repeat
  - evaluations on M “folds”
- Applies to scans/events/blocks/subjects/...
  - Leave-one-out (LOO) approach
Confusion matrix & accuracy

Confusion matrix
  = summary table

Accuracy estimation
  • Class 0 accuracy, $p_0 = \frac{A}{A+B}$
  • Class 1 accuracy, $p_1 = \frac{D}{C+D}$
  • Accuracy, $p = \frac{A+D}{A+B+C+D}$

Other criteria
  • Positive Predictive Value, $PPV = \frac{D}{B+D}$
  • Negative Predictive Value, $NPV = \frac{A}{A+C}$
Accuracy & Dataset balance

Watch out if #samples/class are different!

Example:
Good overall accuracy (72%) but
• Majority class ($N_1 = 80$), excellent accuracy (90%)
• Minority class ($N_2 = 20$), poor accuracy (0%)

Good practice:
Report
• class accuracies $[p_0, p_1, ..., p_C]$
• balanced accuracy $p_{\text{bal}} = (p_0 + p_1 + ... + p_C)/C$
Regression MSE

- LOO error in one fold
  \[ SE_n = (y_n - f(x_n))^2 \]
- Across all LOO folds
  \[ R(f, X) = MSE = \frac{1}{N} \sum_{n=1}^{N} (y_n - f(x_n))^2 \]
  → Out-of-sample “mean squared error” (MSE)

Other measure:
Correlation between predictions (across folds!) and ‘true’ targets
Inference by permutation testing

- $H_0$: “class labels are non-informative”
- Test statistic = CV accuracy
- Estimate distribution of test statistic under $H_0$
  - Random permutation of labels
  - Estimate CV accuracy
  - Repeat M times
- Calculate p-value as

$$\frac{1}{M} \sum_{m=1}^{M} (p_{perm}^m \geq p_{real})$$
Overview

• Introduction
  – Univariate & multivariate approaches
  – Data representation

• Pattern Recognition
  – Machine learning
  – Validation & inference
  – Weight maps & feature selection
  – fMRI application
  – Multiclass problem

• Conclusion & PRoNTo
Weight vector interpretation

Weight vector

- weight (or discrimination) image!
- how important each voxel is
- for which class “it votes” (mean centred data & b=0)

Weight vector

\[ W = \begin{bmatrix} 0.45 & 0.89 \end{bmatrix} \]

\[ b = -2.8 \]
Example of masks

Linear machine → Weight map
Feature selection

- 1 sample image → 1 predicted value
- use ALL the voxels → NO thresholding of weight allowed!

Feature selection:
- a priori mask
- a priori ‘filtering’
- recursive feature elimination/addition → nested cross-validation
  (MUST be independent from test data!)
Overview

• Introduction
  – Univariate & multivariate approaches
  – Data representation

• Pattern Recognition
  – Machine learning
  – Validation & inference
  – Weight maps & feature selection
  – fMRI application
  – Multiclass problem

• Conclusion & PRoNTo
fMRI designs

Level of inference

• within subject ≈ FFX with SPM
  ➔ ‘decode’ subject’s brain state

• between subjects ≈ RFX with SPM
  ➔ ‘classify’ groups, or
    regress subjects’ parameter
Between subjects

Design
• 2 groups: group A vs. group B
• 1 group: 2 conditions per subject

→ Extract 1 (or 2) summary image(s) per subject, and classify

Leave-one-out (LOO) cross-validation:
• Leave one subject out (LOSO)
• Leave one subject per group out (LOSGO)

Note: this works for any type of image...
Within subject

Design:
- Block or event-related design
- Accounting for haemodynamic function

Use single scans

Data Matrix =

<table>
<thead>
<tr>
<th>C1</th>
<th>C1</th>
<th>C1</th>
<th>BL</th>
<th>BL</th>
<th>BL</th>
<th>C2</th>
<th>C2</th>
<th>C2</th>
<th>BL</th>
<th>BL</th>
<th>BL</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
</tr>
</tbody>
</table>

volumes
Within subject

Design:
- Block or event-related design
- Accounting for haemodynamic function
Within subject

Design:
- Block or event-related design
- Accounting for haemodynamic function

Averaging/deconvolution

Data Matrix =

How to?
- Average scans over blocks/events
- Parameter estimate from the GLM with 1 regressor per block/event
Within subject

Design:
- Block or event-related design
- Accounting for haemodynamic function

Leave-one-out (LOO) cross-validation:
- Leave one session/run out
- Leave one block/event out

*(danger of dependent data!!!)*
Overview

• Introduction
  – Univariate & multivariate approaches
  – Data representation

• Pattern Recognition
  – Machine learning
  – Validation & inference
  – Weight maps & feature selection
  – fMRI application
  – Multiclass problem

• Conclusion & PRoNTo
Multiclass problem

<table>
<thead>
<tr>
<th>ECOC</th>
<th>SVM codewords</th>
<th>C1-C2</th>
<th>C1-C3</th>
<th>C2-C3</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td>0</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Example</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>C3</td>
<td></td>
</tr>
</tbody>
</table>

“Error-Correcting Output Coding” (ECOC) approach
Overview

- **Introduction**
  - Univariate & multivariate approaches
  - Data representation

- **Pattern Recognition**
  - Machine learning
  - Validation & inference
  - Weight maps & feature selection
  - fMRI application
  - Multiclass problem

- **Conclusion & PRoNTo**
Conclusions

Key points:

• More sensitivity (~like omnibus test with SPM)
• NO local (voxel/blob) inference
  ➔ CANNOT report coordinates nor thresholded weight map
• Require cross-validation (split in train/test sets)
  ➔ report accuracy/PPV (or MSE)
• MUST assess significance of accuracy
  ➔ permutation approach
PRoNTTo

“Pattern Recognition for Neuroimaging Toolbox”, aka. PRoNTTo:
http://www.mlnl.cs.ucl.ac.uk/pronto/
with references, manual, demo data, course, etc.

Paper: http://dx.doi.org/10.1007/s12021-013-9178-1
Thank you for your attention!

Any question?

Thanks to the PRoNTo Team for the borrowed slides. 😊